

# Preuves

## Sommaire

- Documentation.....2
  - Je sais décrire le contexte de mon application, pour que n’importe qui soit capable de comprendre à quoi elle sert.....2
  - Je sais concevoir et décrire un diagramme de cas d’utilisation pour mettre en avant les différentes fonctionnalités de mon application.....2
  - Je sais concevoir un diagramme UML de qualité représentant mon application.....2
  - Je sais décrire mon diagramme UML en mettant en valeur et en justifiant les éléments essentiels. ....2
- Code.....3
  - Je sais développer en utilisant le SDK le plus bas possible.....3
  - Je sais distinguer mes ressources en utilisant les qualifier.....3
  - Je sais faire des vues xml en utilisant layouts et composants adéquats.....3
  - Je sais coder proprement mes activités, en m’assurant qu’elles ne font que relayer les évènements .....4
  - Je sais coder une application en ayant un véritable métier.....4
  - Je sais utiliser le findViewById à bon escient.....4
  - Je sais coder mon propre adaptateur.....5
  - Je maîtrise l’usage des fragments.....5
  - Je maîtrise l’utilisation de Git.....5
- Application.....6
  - Je sais développer un jeu intégrant une boucle de jeu threadée observable.....6
  - Je sais développer un jeu graphique sans utiliser de SurfaceView.....6

## Documentation

**Je sais décrire le contexte de mon application, pour que n'importe qui soit capable de comprendre à quoi elle sert.**

cf. [contexte](#)

**Je sais concevoir et décrire un diagramme de cas d'utilisation pour mettre en avant les différentes fonctionnalités de mon application.**

cf. [explication du diagramme de cas d'utilisation](#)

**Je sais concevoir un diagramme UML de qualité représentant mon application.**

cf. [diagramme de classes](#)

**Je sais décrire mon diagramme UML en mettant en valeur et en justifiant les éléments essentiels.**

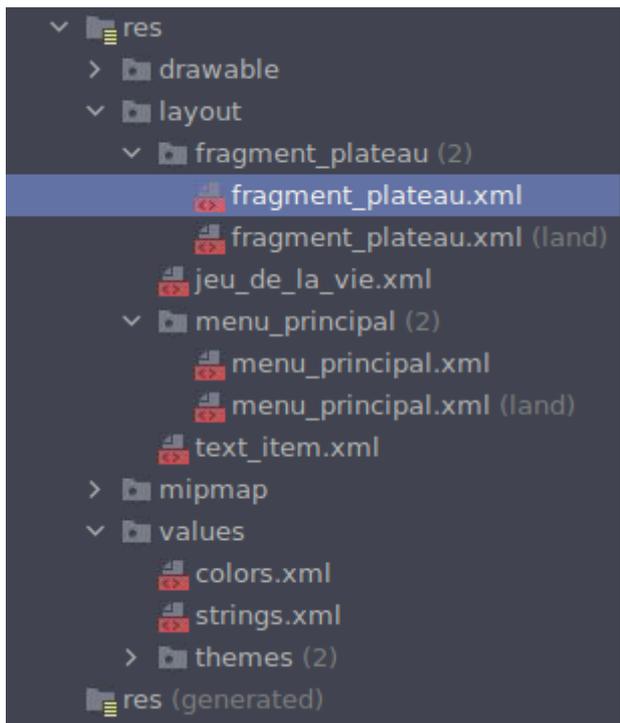
cf. [explication du diagramme de classes](#)

## Code

**Je sais développer en utilisant le SDK le plus bas possible.**

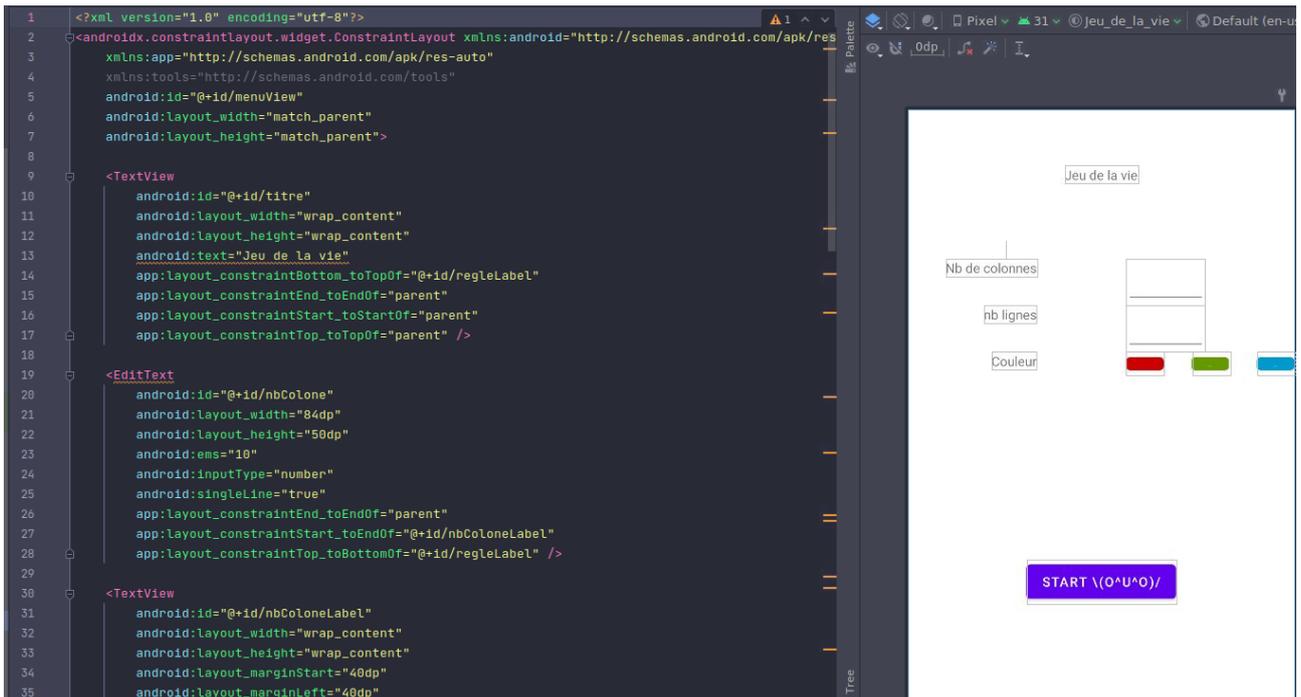
Oui

**Je sais distinguer mes ressources en utilisant les qualifier**



**Je sais faire des vues xml en utilisant layouts et composants adéquats**

cf. [fragment\\_plateau.xml](#), [menu\\_principale.xml](#), [jeu\\_de\\_la\\_vie.xml](#)



**Je sais coder proprement mes activités, en m'assurant qu'elles ne font que relayer les évènements**

cf. [LauncherActivity.java](#)



**Je sais coder une application en ayant un véritable métier**

cf. [Cellule.java](#), [Plateau.java](#), [CellulesVivantes.java](#) ...

**Je sais utiliser le findViewById à bon escient**

cf. [tout les fragments](#)

```
@Override
public void onCreateView(@NonNull View view, @Nullable Bundle savedInstanceState) {
    super.onCreateView(view, savedInstanceState);
    view.findViewById(R.id.startButton).setOnClickListener(l->{startGame(view);});
    Log.d( tag: "fragment", msg: "+view);
}

private int getNbColumnsFromView(View v) throws NumberFormatException{
    EditText editText = v.findViewById(R.id.nbColone);
    String colones = editText.getText().toString();
    Log.d( tag: "fragment", msg: "ICI : "+colones);
    return Integer.parseInt(colones);
}

private int getNbLignesFromView(View v) throws NumberFormatException{
    EditText editText = v.findViewById(R.id.nbLignes);
    String lignes = editText.getText().toString();
    Log.d( tag: "fragment", msg: "ICI : "+lignes);
    return Integer.parseInt(lignes);
}
```

## Je sais coder mon propre adaptateur

cf. [RegleAdaprateur.java](#)

## Je maîtrise l'usage des fragments

cf. [tout les fragments](#), [LauncherActivity.java](#)

## Je maîtrise l'utilisation de Git

cf. [gitlab](#), [git log](#)

## Application

**Je sais développer un jeu intégrant une boucle de jeu threadée observable.**

cf. [les boucles de jeu](#), [ObservableBDJ.java](#), [ObserveurBDJ.java](#), [Manager.java::update](#), [FragmentPlateau.java::update](#)

**Je sais développer un jeu graphique sans utiliser de SurfaceView.**

oui